社団法人 情報処理学会　研究報告
IPSJ SIG Technical Report

2003－CSEC－23　(6)
2003／12／19

簡単で効果的な個人情報保護の方法
— データ保存法とデータ入力法 —

鈴木英男 [1]

あらまし　一般に一個人の個人情報には様々な項目がある. 第三者がその一項目のみの情報を知り得た場合に, その情報に重要性が少ないことに着目し, 各項目を分割して, データを保存したり入力する方法を提案する. 項目間の関連が分からなくなってしまっては, データの活用ができなくなってしまうし, 項目間の関連が容易に推察できるようでは, 分割しても意味がない. これら問題を, ブロック暗号を用いて解決し, 簡単で効果的な個人情報保護の方法を示した. ここで述べた方式は, 個人情報に限らず, 経理データなど様々なデータに適用できるものである.

キーワード: プライバシー, 個人情報, 個人情報保護, ブロック暗号, 暗号

A simple and effective method to protect personal information
— data storage system and data entry system —

Hideo Suzuki [2]

Abstract　In general, one's personal information includes various items. As each item itself is not important, we propose a method that we treat items to separate each other in data storage and data entry systems. We need the relation between items, but the relation should not be resolved easily. To achieve this relation, we adopt a block cipher to connect items. This method will be a simple and effective method to protect personal information. This method can be applied to various data such as company's accounting data.

Keywords: privacy, personal information, personal information protection, block cipher, cryptography.

[1] 東京情報大学 環境情報学科, 〒 265-8501 千葉市若葉区谷当町 1200-2, suzuki@rsch.tuis.ac.jp
[2] Dept. of Environmental Information, Tokyo University of Information Sciences, 1200-2 Yatohcho Wakabaku, Chibashi 265-8501 Japan, suzuki@rsch.tuis.ac.jp

1　まえがき

近年, 個人情報保護の大切さが注目されている. 個人情報を扱う企業や団体は, 個人情報の漏洩があると, 社会的信用を失うことになる. 近年, 企業においては, 一時の利益を得ることよりも, 社会的信用を得ることの大切さが注目されている. 企業が社会的信用を失わないためには, 扱う個人情報の保護が必須事項である.

従来, 個人情報の保護には, 個人情報を扱う人の制限, 扱う人が使用できる権限の制限などを行うために, アクセス制限, 認証, ログ管理などの対策が行われてきた. これら対策は, 非常に効果的である. しかしながら, これら対策をもってしても, 個人情報を書いたカード (はがき) などをキーパンチャがデータ入力する場合, 扱う人は一個人の完全な個人情報を見ることになり問題であった. この対策を 3 節で述べる.

また, 個人情報を扱うデータベースシステムを管理する, 外注コンピュータ業者は, 大量の個人情報を覗きみたり, 盗難することがあり, 大きな社会問題となっている. この対策を 4 節で述べる.

本論文では, これら問題の対策にもなる, 簡単で効果的な個人情報保護の方法 を述べる.

2　個人情報の性質

一般にある一人の個人情報には様々な項目がある. 第三者がその一項目のみの情報を知り得た場合に, その情報に重要性が少ない場合も多い. 本論文ではこの性質に着目し, 種々の対策を考察していく. 図 1 に示すように, 氏名, 性別, 住所, 郵便番号, 電話番号などの個人情報があった場合に, 図 2 のように, 氏名＋性別を項目群 a に, 住所＋郵便番号を項目群 b に, 電話番号を項目群 c に分割すると, 1 つの項目群を知り得ても, 個人情報が成立しない. それならば, データを入力したり保存する場合にも, 各項目群に分割しておけば, 個人情報の漏洩に結びつき難くなる. ただ, 分割したままでは, データの活用ができなくなってしまう. そこで次節では, 項目群間をブロック暗号 AES[1][2] で保護しつつ連結させる, 個人情報の新しいデータ入力法・保存法を述べる.

3　新しいデータ入力法・保存法

まず, 個人情報の具体例を設定する. ビールなどを販売する会社が, シールを集めて, カード (はがき) で応募すると景品が当たるようなキャンペーンを実施していると仮定する. そのビール会社は, キャンペーン集計作業を別会社に委託し, キャンペーン集計会社は, 個人情報の含まれるカード (はがき) のデータ入力を外注データ入力会社に委託しているとする.

この節では, キャンペーン集計会社の立場で, データ入力依頼会社 (ビール会社) からキャンペーンカード (はがき) を受け取り, そのデータを電子ファイル化して, データ入力依頼会社に渡すまでの手順を, 付録 A と付録 B に示す. 方式 A の特徴は, 外注データ入力会社に 13. から 17. の仕事を依頼しても, 一個人の完全な個人情報データを見せることにならないので, 非常に簡単で効果的な個人情報保護が実現できる. この方式のセキュリティ確保のポイントは, カード枚数 n を大きくすることである. n は 1000 以上が望ましい.

この新しいデータ入力法・保存法 (方式A) では, 各項目群の関係を結びつけるシリアル SER を, 外注データ入力会社に隠すために, ブロック暗号 AES256[1][2] で暗号化シリアル ESER に変換している. AES を含む近年のブロック暗号は, 暗号化・復号ソフトウェアを公開しても, 鍵さえ安全に保管しておけば, 暗号文から平文を解読することが困難であるため, 鍵のみを秘密にしておけば良い, という特徴があり, 非常に使いやすいものである.

ただし, 暗号文のセキュリティ確保のためには, 鍵と平文 (暗号文) の bit サイズを大きくしておく必要がある. この結果, 暗号化シリアル ESER は, 288bit と長くなり, 各処理の過程で毎回 288bit のデータがつきまとうことになるというデメリットも存在する. 実は, 方式Aの 14. データ入力で必要なのは, 9. 並べ換えの後の各項目群内での順番だけである. この特徴を基に改良したものが方式 B である.

方式 B の特徴は, 外注データ入力会社に 17. からと 21. の仕事を依頼しても, 一個人の完全な個人情報データを見せることにならないので, 非常に簡単で効果的な個人情報保護が実現できる. しかも, 方式 B は方式 A と比較して, ESER(288 bit) から LIST(64 bit) へと格段に扱うデータ量を減少させることができ, 外注データ入力会社が鍵 K を知り得た場合にも, ESER(288 bit) の内容を見られる心配がないという利点が

ある．この方式Bにおいても, セキュリティ確保のポイントは, カード枚数nを大きくすることである．nは1000以上が望ましい．

方式Aと方式Bで出現する基本SER(176 bit), SER(256 bit), ESER(288 bit), LIST(64 bit) などの書式の構造を図6に示す．

基本SER(176 bit) の構造

入力依頼会社 (16 bit)	カード種別 (24 bit)	日付 (64 bit)	外注入力会社 (8 bit)	区分 (16 bit)	束番号 (24 bit)	束内シリアル番号 (24 bit)

SER(256 bit) の構造

入力依頼会社 (16 bit)	カード種別 (24 bit)	日付 (64 bit)	外注入力会社 (8 bit)	区分 (16 bit)	束番号 (24 bit)	束内シリアル番号 (24 bit)	項目群番号 (16 bit)	予備項目 (0 bit)	乱数R (64 bit)

ESER(288 bit) の構造

暗号化方式コード (8 bit)	予備コード (8 bit)	項目群番号 (16 bit)	暗号化されたSER(基本ESER) (256 bit)

LIST(64 bit) の構造

暗号化方式コード (8 bit)	予備コード (8 bit)	項目群番号 (16 bit)	通し番号 (32 bit)

図6.　基本SER(176 bit), SER(256 bit), ESER(288 bit), LIST(64 bit) の構造

1枚のカードに, 1つの基本SER(176 bit) を用意し, 対応させる．n枚のカードがあると, n個の基本SER(176 bit) がある．例えば, 1枚のカードの各項目を3つの項目群に分割するとすると, 1枚のカードに, 3つのSER(256 bit) を用意し, 対応させる．この場合, n枚のカードがあると, 3n個のSER(256 bit) がある．1つのSER(256 bit) に, 1つのESER(288 bit) を対応させるので, n枚のカードがあると, 3n個のESER(288 bit) がある．1つのESER(288 bit) に, 1つのLIST(64 bit) を対応させるので, n枚のカードがあると, 3n個のLIST(64 bit) がある．

入力依頼会社 (16 bit) とカード種別 (24 bit) は, 将来, 基本SER(176 bit) とSER(256 bit) の構造を変更するときの書式識別コードにもなり得る．ESER(288 bit) には, 暗号化されたSER(基本ESER) を含んでいる．暗号化する前のSERの内容は規則性があり, せっかく暗号化しても暗号文を解読させるきっかけとなり得る可能性があるので, SERには, その規則性を乱す乱数R(64 bit) を入れてある．将来的には, 乱数Rのbit数を減らすことで予備項目のbit数を増やすことができる．乱数Rは各カード, 各項目群ですべて異なる値である．

4　新しいデータ入力法・保存法を活用するための新しいデータベース利用法

付録Cにその方法を示す．この方法は, この個人情報データベースシステムにおいて, 一個人の完全な個人情報データを得るためには, メインメモリを覗くか, USBメモリ＋CDまたはHDDが必要となる．すなわち, USBメモリを外した状態で, CDまたはHDDの中には, 一個人の関連づけられた完全な個人情報データが得られないことになる．これにより, データベースシステムの保守時 (故障時やメンテナンス時) には, USBメモリを外すことで, 保守を別会社に委託しても, 非常に簡単で効果的な個人情報保護が実現できることになる．

5　むすび

本論文では, ビール会社のキャンペーンはがきを例に挙げて, 具体的な個人情報処理手順を示した．これら手順は, 任意の個人情報に利用できる手法である．

最近は, コストを安く上げる目的で, 会社業務の一部を, 外部の専門会社に外注することは珍しくなく, 社外秘のデータを外注で処理することも多い．外注企業と秘守契約を締結しても, リスクは付きものである．リスクを冒したくなければ, 外注せずに自社で処理し費用が上がり, 外注すれば, 費用は安くなるが, リスクが発生してしまうトレードオフの関係があった．

そこで, 本論文の各方式を実行すれば, リスクを最小限にし, 外注することができ, 上記のようなトレードオフの心配もしなくてよくなる. しかも, 扱えるデータは個人情報に限らず, 経理データなど様々なデータに適用できるので, データ入力法・保存法の新しい方向性を与えるものである.

参考文献

[1] Joan Daemen and Vincent Rijmen: *The Design of Rijndael*, Springer-Verlag, New York, Jan. 2002.

[2] Advanced Encryption Standard, FIPS-197 `http://csrc.nist.gov/CryptoToolkit/aes/`

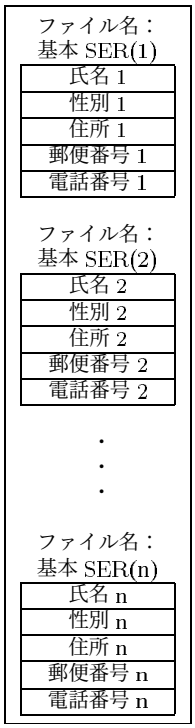

図 1

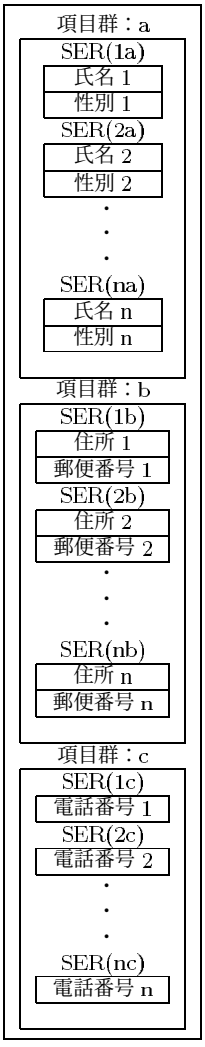

図 2

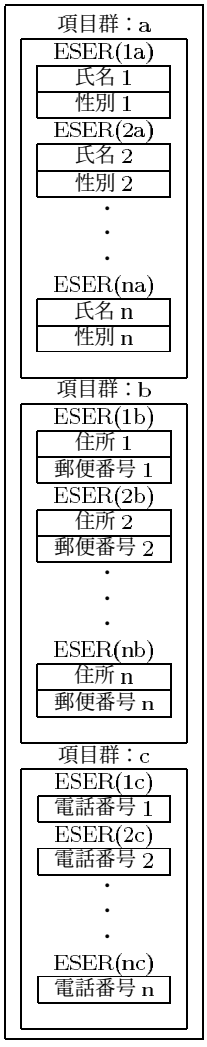

図 3

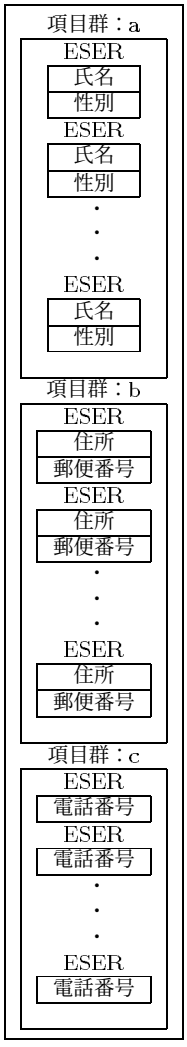

図 4

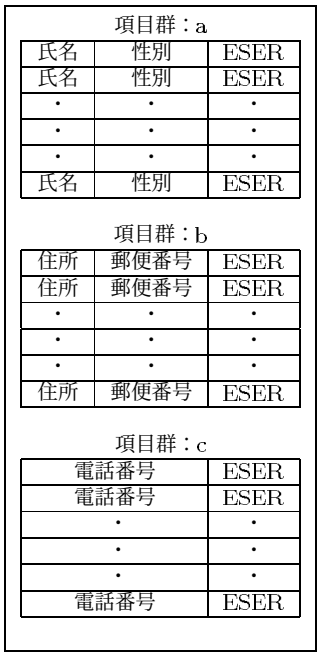

図 5

付録 A：新しいデータ入力法・保存法の手順 (方式 A)

1. データ入力依頼会社から n 枚のカード (はがき) を受け取る.

↓

2. 8. 暗号化で使う鍵 K(256 bit) を用意.

↓

3. 19. 暗号化で使う鍵 K2(256 bit) を用意. (鍵 K2 の数は, 項目群の数だけ異なるものを用意.)

↓

4. 16. 暗号化で使う鍵 K3(256 bit) を用意. (鍵 K3 の数は, 項目群の数だけ異なるものを用意.)

↓

5. n 枚のカード (はがき) をスキャナで読みとり, 画像ファイルに変換する. (n 枚のカードから, n 個のファイルが生成される.)

↓

6. 各画像ファイルに, ファイル名として基本シリアル SER(176 bit) を割り当てる (図 1, 図 6).

↓

7. 各画像ファイルを項目群毎に切り取り, 項目群毎に別の画像ファイルに分割する. 分割後のファイル名としてシリアル SER(256 bit) を割り当てる (図 2, 図 6). (説明を簡単にするため 3 つの項目群に分割すると仮定すると, n 個の画像ファイルから 3n 個の画像ファイルが生成される. 例えば, 氏名の項目と性別の項目を項目群 1, 住所の項目と郵便番号の項目を項目群 2, 電話番号の項目を項目群 3 とすると, 項目群は 3 つとなる.)

↓

8. 3n 個すべての画像ファイルについて, ファイル名を, シリアル SER(256 bit) から暗号化シリアル ESER(288 bit) に変更する (図 3, 図 6). シリアル SER(256 bit) の暗号化には, 鍵 K(256 bit) を用いてブロック暗号 AES256 により暗号化する.

↓

9. この時点では, 項目群毎の各ファイルは, 5. スキャナで読み取ったときの順番に並んでいる可能性があるので, 乱数にも見える暗号化シリアル ESER(288 bit) の値をキーとして並べ換え (ソート) を行う (図 4).

↓

10. 並べ替えを行っても, 各ファイルのタイムスタンプは, カード順になっている可能性があるので, 3n 個すべての画像ファイルを同じ日付, 同じ時間に変更する. 変更後の日付は, シリアル番号 SER 内の日付と同じでも良い.

↓

11. 以上で生成された, 暗号化シリアル ESER(288 bit) をファイル名としてもつ, 3n 個の画像ファイルを, VPN(virtual private network) などの安全な回線で, 外注データ入力会社のサーバに転送 (送信) する.

↓

12. 3 個の鍵 K3(256 bit) を, VPN(virtual private network) などの安全な回線で, 外注データ入力会社のサーバに転送 (送信) する.

↓

13. 外注データ入力会社では, 暗号化シリアル ESER(288 bit) をファイル名としてもつ, 3n 個の画像ファイルを受信する.

↓

14. キーパンチャがデータ入力することで, 画像情報をテキストデータへと変換し 3n 個のテキストファイルを生成する. 各画像ファイルに対応するテキストファイルのファイル名も画像ファイルと同じ暗号化シリアル ESER(288 bit) となる.

↓

15. 外注データ入力会社のサーバでは, 各項目群で n 個あるファイルを, 1 つのファイルに集約する. すなわち, 3n 個のファイルを項目群でまとめて, 3 個のファイルに変換する (図 5).

↓

16. 15. で変換された 3 つのファイルは, それぞれ, 異なる 3 個の鍵 K3(256 bit) により, ブロック暗号 AES256 により, 暗号化する. (15. と 16. の処理は, 完全にサーバの中で閉じて行うこととする.)

↓

17. 以上で生成された 3 個のファイルを, VPN(virtual private network) などの安全な回線で, 外注データ入力会社から当社のサーバに転送 (送信) する.

↓

18. 暗号化された 3 つのファイルは, それぞれ, 異なる 3 個の鍵 K3(256 bit) により, ブロック暗号 AES256 により, 復号する.

↓

19. 復号された 3 つのファイルは, それぞれ, 異なる 3 個の鍵 K2(256 bit) により, ブロック暗号 AES256 により, 暗号化する.

↓

20. 3 つの暗号化ファイルを CD-R などに保存する.

↓

21. 鍵 K(256 bit), 3 個の鍵 K2(256 bit) と CD-R をデータ入力依頼会社に納める.

付録 C：新しいデータ入力法・保存法を活用するための新しいデータベース利用法の手順

1. 個人情報管理者は鍵 K, 鍵 K2 を USB メモリに書き込む.

↓

2. 新しいデータ入力法・保存法で得られた 3 つの暗号化ファイルを準備する (CD または HDD 内).

↓

3. 3 つの暗号化ファイルを, 鍵 K2 により, ブロック暗号 AES256 により, 復号する (メインメモリ内で作業する).

↓

4. 復号された 3 つのファイル中の 288bit 暗号化シリアル番号 ESER を, 鍵 K により, ブロック暗号 AES256 により, 復号し, 256bit シリアル番号 SER に変換する (メインメモリ内で作業する).

↓

5. 同じ SER をもつデータを結合し, テーブルを作成し, 保存する (メインメモリ内).

付録B：新しいデータ入力法・保存法の手順(方式B)

1. データ入力依頼会社から n 枚のカード (はがき) を受け取る.

↓

2. 8. 暗号化で使う鍵 K(256 bit) を用意.

↓

3. 19. 暗号化で使う鍵 K2(256 bit) を用意. (鍵 K2 の数は, 項目群の数だけ異なるものを用意.)

↓

4. 16. 暗号化で使う鍵 K3(256 bit) を用意. (鍵 K3 の数は, 項目群の数だけ異なるものを用意.)

↓

5. 項目群毎の各画像ファイル用に, シリアル SER(256 bit) を準備する. (項目群の数は 12. で決められる. 6. 以後の処理で必要となるので先取りして, 3 つの項目群に分割すると仮定する.)

↓

6. 3n 個のシリアル SER(256 bit) から 3n 個の暗号化シリアル ESER(288 bit) を計算し, 3 個の SER(256 bit) ←→ ESER(288 bit) 対応表 (n 行 2 列) を作成する. シリアル SER(256 bit) の暗号化には, 鍵 K(256 bit) を用いてブロック暗号 AES256 により暗号化する.

↓

7. それぞれの SER(256 bit) ←→ ESER(288 bit) 対応表 (n 行 2 列) を乱数にも見える暗号化シリアル ESER(288 bit) の値をキーとして表の並べ換え (ソート) を行う.

↓

8. 7. でソートされた表の 1 行目～n 行目の各行に通し番号を付け, その項目をリスト番号 LIST(64 bit) とする. すなわち, 3 個の SER(256 bit) ←→ ESER(288 bit) ←→ LIST(64 bit) 対応表 (n 行 3 列) を作成する.

↓

9. LIST(64 bit) のうち, 上位 32bit には, ESER と同じ暗号化方式コード (8 bit), 予備コード (8 bit), 項目群番号 (16 bit) を書き込む. 10. それぞれの SER(256 bit) ←→ ESER(288 bit) ←→ LIST(64 bit) 対応表 (n 行 3 列) を SER(256 bit) の値をキーとして表の並べ換え (ソート) を行う.

↓

11. n 枚のカード (はがき) をスキャナで読みとり, 画像ファイルに変換する. (n 枚のカードから, n 個のファイルが生成される.)

↓

12. 各画像ファイルを項目群毎に切り取り, 項目群毎に別の画像ファイルに分割する. 分割後のファイル名としてリスト番号 LIST(64 bit) を割り当てる. (説明を簡単にするため 3 つの項目群に分割すると仮定すると, n 個の画像ファイルから 3n 個の画像ファイルが生成される. 例えば, 氏名の項目と性別の項目を項目群 1, 住所の項目と郵便番号の項目を項目群 2, 電話番号の項目を項目群 3 とすると, 項目群は 3 つとなる.)

↓

13. この時点では, 項目群毎の各ファイルは, 11. スキャナで読み取ったときの順番に並んでいる可能性があるので, 乱数にも見えるリスト番号 LIST(64 bit) の値をキーとして並べ換え (ソート) を行う.

↓

14. 並べ替えを行っても, 各ファイルのタイムスタンプは, カード順になっている可能性があるので, 3n 個すべての画像ファイルを同じ日付, 同じ時間に変更する.

↓

15. 以上で生成された, リスト番号 LIST(64 bit) をファイル名としてもつ, 3n 個の画像ファイルを, VPN(virtual private network) などの安全な回線で, 外注データ入力会社のサーバに転送 (送信) する.

↓

16. 3 個の鍵 K3(256 bit) を, VPN(virtual private network) などの安全な回線で, 外注データ入力会社のサーバに転送 (送信) する.

↓

17. 外注データ入力会社では, リスト番号 LIST(64 bit) をファイル名としてもつ, 3n 個の画像ファイルを受信する.

↓

18. キーパンチャがデータ入力することで, 画像情報をテキストデータへと変換し 3n 個のテキストファイルを生成する. 各画像ファイルに対応するテキストファイルのファイル名も画像ファイルと同じリスト番号 LIST(64 bit) となる.

↓

19. 外注データ入力会社のサーバでは, 各項目群で n 個あるファイルを, 1 つのファイルに集約する. すなわち, 3n 個のファイルを項目群でまとめて, 3 個のファイルに変換する.

↓

20. 19. で変換された 3 つのファイルは, それぞれ, 異なる 3 個の鍵 K3(256 bit) により, ブロック暗号 AES256 により, 暗号化する. (19. と 20. の処理は, 完全にサーバの中で閉じて行うこととする.)

↓

21. 以上で生成された 3 個のファイルを, VPN(virtual private network) などの安全な回線で, 外注データ入力会社から当社のサーバに転送 (送信) する.

↓

22. 暗号化された 3 つのファイルは, それぞれ, 異なる 3 個の鍵 K3(256 bit) により, ブロック暗号 AES256 により, 復号する.

↓

23. 10. で得られた 3 つの SER(256 bit) ←→ ESER(288 bit) ←→ LIST(64 bit) 対応表 (n 行 3 列) により, 3 つのファイル中の LIST(64 bit) を対応する ESER(288 bit) に変換する.

↓

24. 変換された 3 つのファイルは, それぞれ, 異なる 3 個の鍵 K2(256 bit) により, ブロック暗号 AES256 により, 暗号化する.

↓

25. 3 つの暗号化ファイルを CD-R などに保存する.

↓

26. 鍵 K(256 bit), 3 個の鍵 K2(256 bit) と CD-R をデータ入力依頼会社に納める.